

NETWORK PRINTER AND NETWORK PRINTING SYSTEM

Patent Number: JP2001067195
Publication date: 2001-03-16
Inventor(s): JINGU HIDETO; IGUCHI MANABU
Applicant(s): OKI DATA CORP
Requested Patent: ☐ JP2001067195
Application Number: JP19990239088 19990826
Priority Number(s):
IPC Classification: G06F3/12; B41J29/38
EC Classification:
Equivalents:

Abstract

PROBLEM TO BE SOLVED: To efficiently process a printing job even when a printer on a network can not execute printing operation.

SOLUTION: Protocols supported by other printers and priority information are stored in a subprinter information storage part 7. When the printer itself is in a printing disabled state, a protocol control part 8 determines a substitutive printer for the printer itself on the basis of the priority stored in the storage part 7 and converts printing data into a protocol supported by the substitutive printer. A data transmission part 5 transmits the printing data whose protocol is converted by the control part 8 to the substitutive printer.

Data supplied from the esp@cenet database - I2

【特許請求の範囲】

【請求項 1】 ネットワークに接続されている自プリンタとは異なる他のプリンタのサポートしているプロトコルと予め決められた優先順位の情報を格納するサブプリンタ情報格納部と、

前記自プリンタで印刷不可能な状態であった場合、前記サブプリンタ情報格納部の優先順位に基づいて、前記他のプリンタの中から当該自プリンタの代わりとなるプリンタを決定し、前記自プリンタへの印刷データを、前記自プリンタの代わりとなるプリンタのサポートしているプロトコルに変換するプロトコル制御部と、
前記プロトコル制御部で変換した印刷データを、前記代わりとなるプリンタに送信するデータ送信部とを備えたことを特徴とするネットワークプリンタ。

【請求項 2】 請求項 1 に記載のネットワークプリンタにおいて、

自プリンタとは異なる他のプリンタのサポートしているエミュレーションの情報を格納するサブプリンタ情報格納部と、

前記自プリンタで印刷不可能な状態であった場合、プロトコル制御部で決定した前記自プリンタの代わりとなるプリンタのエミュレーションに変換するエミュレーション制御部と、

前記エミュレーション制御部でエミュレーションを変換した印刷データを、前記自プリンタの代わりとなるプリンタに送信するデータ送信部とを備えたことを特徴とするネットワークプリンタ。

【請求項 3】 ネットワーク上のファイルサーバを検索するファイルサーバ検索部と、

自プリンタで印刷不可能な状態であった場合、当該自プリンタへの印刷データを、前記ファイルサーバ検索部で検索したファイルサーバ上にテンポラリデータとして保存し、前記自プリンタが印刷可能な状態に戻った場合に前記ファイルサーバから前記テンポラリデータを自プリンタで印刷するためのデータとして読み出すファイルロード部とを備えたことを特徴とするネットワークプリンタ。

【請求項 4】 ネットワーク上を流れる印刷に使用される情報を印刷ログとして収集する印刷ログ収集部と、

自プリンタで印刷不可能な状態であった場合、前記印刷ログ収集部で収集した印刷ログに基づき、当該自プリンタ以外のプリンタから次の印刷先プリンタとしてふさわしいプリンタを検索する印刷ログ解析部と、
前記印刷ログ解析部で検索したプリンタに対して当該自プリンタへの印刷データを転送するデータ送信部とを備えたことを特徴とするネットワークプリンタ。

【請求項 5】 請求項 4 に記載のネットワークプリンタにおいて、

ネットワーク上を流れる印刷要求元装置の電子メールアドレスの情報を収集する電子メール送信ログ収集部と、

他プリンタに対して印刷データを転送した場合、前記印刷要求元装置の電子メールアドレスに対して、転送した旨の電子メールを送信する電子メール送信ログ解析部とを備えたことを特徴とするネットワークプリンタ。

【請求項 6】 ネットワークを介して印刷要求元装置からネットワークプリンタに印刷を要求するネットワーク印刷システムにおいて、

前記ネットワークプリンタは、

前記ネットワーク上を流れる印刷に使用される情報を印刷ログとして収集する印刷ログ収集部と、

自プリンタで印刷不可能な状態であった場合、前記印刷ログ収集部で収集した印刷ログに基づき、当該自プリンタ以外のプリンタから次の印刷先プリンタとしてふさわしいプリンタを検索する印刷ログ解析部と、

前記印刷ログ解析部で検索したプリンタに対して当該自プリンタへの印刷データを転送するデータ送信部と、

前記データ送信部から印刷データを送信したプリンタの情報を、ポップアップ・メッセージとして前記印刷要求元装置に送るポップアップ・メッセージ送信部とを備え、

前記印刷要求元装置は、
前記ネットワークプリンタから送信されたポップアップ・メッセージを受信するポップアップ・メッセージ受信部と、

前記ポップアップ・メッセージ受信部で受信したポップアップ・メッセージを表示するポップアップ・メッセージ制御部とを備えたことを特徴とするネットワーク印刷システム。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、ネットワークに接続されたネットワークプリンタ及びネットワーク印刷システムに関し、特に、印刷不能時の制御に関するものである。

【0002】

【従来の技術】従来、ネットワークプリンタでの印字を行う場合には次のように行っていた。即ち、各ネットワークの OS の設定方法に従い、印字を行う複数のネットワークプリンタを予めネットワークの管理を行うサーバに登録しておく。そして、クライアント上のアプリケーションから印字を行う段階で、印字を行わせたいネットワークプリンタを指定するデータを含んだジョブをサーバに送信する。すると、そのジョブをサーバが適切なネットワークプリンタに転送し、それによってクライアント側で指定したネットワークプリンタに対する印字を行わせることができる。

【0003】また、例えば、Windows 95（マイクロソフト社の登録商標）のピア・ツー・ピア（peer-to-peer）印刷の場合には、セントロニクスインタフェースで接続しているプリンタと同様な方法で、クライアン

ト側で予めネットワークプリンタの登録を行っておき、印字する時にジョブを送信するプリンタを直接指定することによって、適切なネットワークプリンタに対する印字を行わせることができる。

【0004】更に、上記の技術に加えて、複数のネットワークプリンタに対して印字データを自動的に分散させて効率よく印字データを処理する方法として、Novell社のNDPS (NetWare Distributed Print Services) を代表とするいくつかの方法が存在する。その技術を用いることによって、ユーザはその時点でプリンタに存在するジョブの処理状態を知ることなしに印字の効率化を行うことができる。特に、NDPSは接続しているユーザが接続している各ネットワークプリンタに対する設定を行うことなしに、非常に効率のよい印刷環境を提供する。

【0005】

【発明が解決しようとする課題】しかしながら、従来では以下のような問題があった。

【0006】1. 各効率化は、使用されているネットワークOSに依存するものであり、そのため、各ネットワークOSでの印刷効率化には、そのOSがインストールされたサーバの存在が不可欠である。

【0007】2. 印刷先となるプリンタが、どのような通信プロトコルをサポートしているかを印刷するユーザが知らない場合には、印刷元からプリンタに対して、プリンタ側で受信不可能なデータ形式で送信してしまい、プリンタが印字不可となってしまう場合があった。

【0008】3. 印刷先となるプリンタが、どのようなエミュレーションをサポートしているかを印刷するユーザが知らない場合には、印刷元からプリンタに対して、プリンタ側で印刷不可能な印字データ形式で送信してしまい、プリンタが未サポートの印字データを印字しようとするため、多量のゴミ印字を出力してしまう場合があった。

【0009】4. 印刷先プリンタが用紙切れや紙詰まり等の印字不可能な状態になってしまった場合に、その障害を取り除くまでは、それ以降のジョブを処理することができなかった。

【0010】

【課題を解決するための手段】本発明は、前述の課題を解決するため次の構成を採用する。

〈構成1〉ネットワークに接続されている自プリンタとは異なる他のプリンタのサポートしているプロトコルと予め決められた優先順位の情報を格納するサブプリンタ情報格納部と、自プリンタで印刷不可能な状態であった場合、サブプリンタ情報格納部の優先順位に基づいて、他のプリンタの中から自プリンタの代わりとなるプリンタを決定し、自プリンタへの印刷データを、自プリンタの代わりとなるプリンタのサポートしているプロトコルに変換するプロトコル制御部と、プロトコル制御部で変

換した印刷データを、代わりとなるプリンタに送信するデータ送信部とを備えたことを特徴とするネットワークプリンタ。

【0011】〈構成2〉構成1に記載のネットワークプリンタにおいて、自プリンタとは異なる他のプリンタのサポートしているエミュレーションの情報を格納するサブプリンタ情報格納部と、自プリンタで印刷不可能な状態であった場合、プロトコル制御部で決定した自プリンタの代わりとなるプリンタのエミュレーションに変換するエミュレーション制御部と、エミュレーション制御部でエミュレーションを変換した印刷データを、自プリンタの代わりとなるプリンタに送信するデータ送信部とを備えたことを特徴とするネットワークプリンタ。

【0012】〈構成3〉ネットワーク上のファイルサーバを検索するファイルサーバ検索部と、自プリンタで印刷不可能な状態であった場合、自プリンタへの印刷データを、ファイルサーバ検索部で検索したファイルサーバ上にテンポラリデータとして保存し、自プリンタが印刷可能な状態に戻った場合にファイルサーバからテンポラリデータを自プリンタで印刷するためのデータとして読み出すファイルロード部とを備えたことを特徴とするネットワークプリンタ。

【0013】〈構成4〉ネットワーク上を流れる印刷に使用される情報を印刷ログとして収集する印刷ログ収集部と、自プリンタで印刷不可能な状態であった場合、印刷ログ収集部で収集した印刷ログに基づき、自プリンタ以外のプリンタから次の印刷先プリンタとしてふさわしいプリンタを検索する印刷ログ解析部と、印刷ログ解析部で検索したプリンタに対して自プリンタへの印刷データを転送するデータ送信部とを備えたことを特徴とするネットワークプリンタ。

【0014】〈構成5〉構成4に記載のネットワークプリンタにおいて、ネットワーク上を流れる印刷要求元装置の電子メールアドレスの情報を収集する電子メール送信ログ収集部と、他プリンタに対して印刷データを転送した場合、印刷要求元装置の電子メールアドレスに対して、転送した旨の電子メールを送信する電子メール送信ログ解析部とを備えたことを特徴とするネットワークプリンタ。

【0015】〈構成6〉ネットワークを介して印刷要求元装置からネットワークプリンタに印刷を要求するネットワーク印刷システムにおいて、ネットワークプリンタは、ネットワーク上を流れる印刷に使用される情報を印刷ログとして収集する印刷ログ収集部と、自プリンタで印刷不可能な状態であった場合、印刷ログ収集部で収集した印刷ログに基づき、自プリンタ以外のプリンタから次の印刷先プリンタとしてふさわしいプリンタを検索する印刷ログ解析部と、印刷ログ解析部で検索したプリンタに対して自プリンタへの印刷データを転送するデータ送信部と、データ送信部から印刷データを送信したプリ

ンタの情報を、ポップアップ・メッセージとして印刷要求元装置に送るポップアップ・メッセージ送信部とを備え、印刷要求元装置は、ネットワークプリンタから送信されたポップアップ・メッセージを受信するポップアップ・メッセージ受信部と、ポップアップ・メッセージ受信部で受信したポップアップ・メッセージを表示するポップアップ・メッセージ制御部とを備えたことを特徴とするネットワーク印刷システム。

【0016】

【発明の実施の形態】以下、本発明の実施の形態を具体例を用いて詳細に説明する。

《具体例1》具体例1では、クライアントが印字先として指定したネットワークプリンタにおいて、何らかの原因で印字不可能なエラーが発生した場合、受信した印字データを、予め設定されている他のネットワークプリンタ（サブプリンタと呼ぶ）に再転送して印字するものである。その際、本具体例が組み込まれたプリンタは、サブプリンタがサポートしている通信プロトコルに変換した後、データを送信する。

【0017】〈構成〉図1は本発明のネットワークプリンタの具体例1を示す構成図である。図の装置は、コントローラ部1とエンジン部2からなる。コントローラ部1は、ネットワークプリンタとしての各種の制御を行う機能を有し、エンジン部2は印刷データを実際に印刷する機能を有している。

【0018】コントローラ部1は、エミュレーション制御部3、データ受信部4、データ送信部5、MIBデータアクセス処理部6、サブプリンタ情報格納部7、プロトコル制御部8からなる。また、プロトコル制御部8は、プロトコル解析部9とプロトコル変換部10からなる。

【0019】エミュレーション制御部3は、印刷データの編集を行い、エンジン部2への印刷データとして出力する機能を有している。データ受信部4は、印刷データをネットワークを介して受信する機能を有している。データ送信部5は、サブプリンタに対してデータを再送信する機能を有している。MIBデータアクセス処理部6は、サブプリンタ情報格納部7へのアクセスを行う機能を有している。サブプリンタ情報格納部7は、サブプリンタに対しての情報を格納する機能部である。

【0020】図2は、サブプリンタ情報格納部7の内容例を示す説明図である。図示の内容例は、各サブプリンタにデータの転送を行うための優先順位、プリンタ名、MACアドレス、サポートしているプロトコル、サポートしているエミュレーションの情報からなる。これらのデータの格納方法の一つとして、MIB (Manage Information Base) にサブプリンタ情報という独自のオブジェクトを定義し、プリンタの情報の一つとして保持する方法で実現する。

【0021】サブプリンタ情報における一つのサブプリ

ンタに対するデータの項目として、“優先順位”、“プリンタ名”、“MACアドレス”、“サポートしているプロトコル”、“サポートしているエミュレーション”があり、“サポートしているプロトコル”と“サポートしているエミュレーション”については、複数のデータである場合がある。例えば、最も優先順位の高いサブプリンタのプリンタ名は“PR2”であり、そのMACアドレスは“00:80:87:XX:XX:XX”である。また、このプリンタは通信プロトコルとして“LPR”と“FTP”をサポートしており、印刷エミュレーションとして“PCL”、“IBM PPR”、“EPSON FX”をサポートしていることを示している。尚、図2中のこれらプロトコルやエミュレーションの名称は、一般に登録商標または商標である。

【0022】図1に戻って、プロトコル制御部8は、自プリンタで印刷不可能な状態であった場合、サブプリンタ情報格納部7の優先順位に基づいて、他のプリンタの中から自プリンタの代わりとなるプリンタを決定し、自プリンタへの印刷データをそのプリンタがサポートしているプロトコルに変換する機能を有するものである。

【0023】また、プロトコル解析部9は、データ受信部4で受信したデータのプロトコルを解析し、印刷データか制御情報かの判断を行って、印刷データであった場合はエンジン部2のステータス情報2aに基づき、印刷データをエミュレーション制御部3に通知するか、プロトコル変換部10を起動するかを判定する機能を有している。即ち、自プリンタで印刷不可能な状態であった場合にプロトコル変換部10を起動するよう制御を行う機能を有している。

【0024】プロトコル変換部10は、プロトコル解析部9によって起動されると、サブプリンタ情報格納部7の優先順位に基づいて、他のプリンタの中から自プリンタの代わりとなるプリンタを決定し、自プリンタへの印刷データをそのプリンタがサポートしているプロトコルに変換する機能を有するものである。

【0025】また、エンジン部2のステータス情報2aは、そのネットワークプリンタが印刷可能であるか印刷不可の状態であるかを示す情報である。

【0026】〈動作〉まず、本装置の動作環境について説明する。図3は、ネットワークプリンタの動作環境を示す説明図である。PR1が本ネットワークプリンタであり、PR2がサブプリンタである。また、PC1が印刷要求元装置である印刷元PCであり、PC2はその他のパソコンを示している。そして、PC1が印刷先プリンタ選択画面でPR1を選択した場合、印刷データがPR1に送られ、PR1はこの印刷データを受け取ると印刷を行う。

【0027】次に、動作の前提となるサブプリンタ情報格納部7の設定方法について説明する。図4は、サブプリンタ情報格納部7の情報設定動作のフローチャートである。まず、プリンタに対してホストからのSNMP (Simple Network Management Protocol) のSet Requestパケ

ットが送信される (ステップ S 1)。

【0028】図5は、ホストからのサブプリンタ情報設定要求の説明図である。ここで、SNMPとは、TCP/IPのネットワーク管理プロトコルであり、ルータ、ハブなどのネットワーク機器 (エージェント) のネットワーク管理情報を管理システム (マネージャ) に送る際の標準プロトコルとして採用されているものである。

【0029】図示のように、このSNMPパケットに含まれる情報としては、プリンタ名やMACアドレスおよびそのプリンタの優先順位、そしてサポートしているプロトコルやエミュレーションといった情報である。

【0030】図4に戻って、プリンタ側では、ホストからのSNMPセットリクエストパケットを受信すると、そのパケットのデータ部から値を読み出し、各設定値をMIB (サブプリンタ情報格納部7) に格納する。

【0031】図6は、プリンタにおけるサブプリンタ情報格納部7への情報格納処理の説明図である。図示のように、優先順位やプリンタ名といった各設定値をMIBに格納する。

【0032】次に、ネットワークプリンタの印刷動作を説明する。図7は、印刷動作時のフローチャートである。まず、ホストから送信されたデータをデータ受信部4が受信バッファ4aに格納し (ステップS11)、これを、プロトコル制御部8におけるプロトコル解析部9に通知する。プロトコル解析部9はプロトコルの解析を行い、このプロトコル情報をMIBデータアクセス処理部6がサブプリンタ情報格納部7に格納した後、プロトコル解析部9は制御情報か印刷データかの判断処理を行う (ステップS12)。ここで、制御情報であった場合は印刷動作とは直接関係がないため、その説明は省略する。

【0033】プロトコル解析部9は、印刷データであった場合、エンジン部2が格納しているステータス情報2aが印刷状態であるか否かを判断する (ステップS13)。ステータス情報2aが印刷不可能状態であった場合、プロトコル解析部9によってプロトコル変換部10が起動され、プロトコル変換処理が行われる (ステップS14)。

【0034】図8は、プロトコル変換処理のフローチャートである。まず、サブプリンタ情報格納部7から優先順位の最も高いサブプリンタを選択し、 $i = 1$ とする (ステップS141)。次に、サブプリンタ情報格納部7に登録されている i 番目のサポートしているプロトコルの情報をリードする (ステップS142)。ここでデータが存在しない場合は、プロトコル解析部9で調査した受信データで使用しているプロトコルは、選択したサブプリンタでサポートされないため、サブプリンタがサポートしているプロトコルの1番目の情報をリードし (ステップS143)、受信データをリードしたプロトコルに変換した (ステップS144) 後、データ送信部

5に通知する (ステップS145)。

【0035】一方、上記ステップS142において、データが存在する場合には、リードしたデータと、受信データで使用されているプロトコルとの比較を行う (ステップS146)。比較の結果が一致していれば、受信データで使用したプロトコルは、選択したサブプリンタでサポートしているため、変換を行わずにデータ送信部5に通知する (ステップS145)。

【0036】ステップS146の比較の結果、不一致の場合には $i \leftarrow i + 1$ とし (ステップS147)、ステップS142に戻って次のサポートしているプロトコルの情報をリードする。

【0037】図9は、プロトコル変換の説明図である。この例では、LPRパケットからFTPパケットへの変換を行っている。この場合に必要なのは、

①ホストと印刷先プリンタを示していたMACアドレス、IPアドレスを、印刷先プリンタとサブプリンタのMACアドレス、IPアドレスに変換すること

②TCP層でのポート番号を、608 (LPR) から1033 (FTP) に変更することである。

【0038】そこで、LPRパケットでは、(PC→PR1) であったMACアドレスを (PR1→PR2) とし、IPアドレス (PC→PR1) を (PR1→PR2) に、TCP層のポート番号を608から1033に変更している。このようにして、本具体例では、サブプリンタがサポートしている通信プロトコルへの変換を行っている。

【0039】また、図7において、ステップS14でプロトコル変換を行った後は、データ送信部5によって、サブプリンタに印刷データを送信する (ステップS15)。一方、ステップS13で印刷が可能であった場合は、通常動作であるため、エンジン部2で印刷可能なデータに変換し (ステップS16)、印刷を行う (ステップS17)。

【0040】そして、自ネットワークプリンタでの印刷不可能時、データ送信部5は、選択したサブプリンタのMACアドレスに対してデータを再送信する。図10は、印刷不可時の動作説明図である。図示のように、本具体例では、PR1に何らかの印刷不可エラーが発生した場合、サブプリンタであるPR2に印刷データを転送する。また、印刷元であるホスト (PC1) に対しては、印刷データ転送先通知メッセージを送信する。

【0041】〈効果〉以上のように具体例1によれば、自プリンタで印刷不可能な場合に印刷データを転送するサブプリンタのプロトコルの情報を予め登録し、印刷不可能な場合は、サブプリンタのプロトコルに変換して転送するようにしたので、ホストから印字先として指定したネットワークプリンタでエラーが発生し、印刷が不可能な場合でもそのネットワークプリンタの資源を最大限に利用し、効率よく印字ジョブを処理することができる。

【0042】《具体例2》具体例2では、上記具体例1のように、クライアントが印字先として指定したネットワークプリンタにおいて、何らかの原因で印字不可能なエラーが発生した場合、サブプリンタの通信プロトコルに変換すると共に、そのエミュレーション（データ形式）に変換するようにしたものである。

【0043】〈構成〉図11は、本発明のネットワークプリンタの具体例2を示す構成図である。図において、具体例1と異なる構成は、コントローラ部1aのエミュレーション制御部3におけるエミュレーション解析部11とエミュレーション変換部12である。エミュレーション解析部11は、印刷データが用いているエミュレーションを解析する機能を有し、エミュレーション変換部12は、サブプリンタがサポートしているエミュレーションに合わせてエミュレーション変換を行う機能を有している。他の各構成については、対応する部分に具体例1と同一の符号を付してその説明を省略する。

【0044】〈動作〉ネットワークプリンタの動作環境やサブプリンタ情報格納部7の情報設定動作は具体例1と同様であるため、印刷動作についてのみ説明する。

【0045】図12は、印刷動作時のフローチャートである。まず、ホストから送信されたデータをデータ受信部4が受信バッファ4aに格納し（ステップS21）、これを、プロトコル制御部8におけるプロトコル解析部9に通知する。プロトコル解析部9はプロトコルの解析を行いプロトコル情報を格納した後、制御情報か印刷データかの判断処理を行う（ステップS22）。

【0046】印刷データであった場合、エンジン部2が格納しているステータス情報2aが印刷状態であるか否かを判断する（ステップS23）。プリンタのステータスが印刷不可能状態の場合、プロトコル解析部9によってプロトコル変換部10が起動され、プロトコル変換処理が行われる（ステップS24）。以上の処理は具体例1と同様である。

【0047】ステップS24のプロトコル変換処理が終了すると、次にエミュレーション制御部3によって、エミュレーション解析部11が起動され、エミュレーション解析が行われる（ステップS25）。エミュレーション解析部11は、受信した印字データが、どのようなエミュレーションを用いているかを解析し、その情報を格納する。次に、エミュレーション制御部3によって、エミュレーション変換部12が起動され、エミュレーション変換が行われる（ステップS26）。

【0048】図13は、エミュレーション変換部12の動作フローチャートである。まず、サブプリンタ情報格納部7から優先順位の最も高いサブプリンタを選択し、 $i = 1$ とする（ステップS261）。次に、サブプリンタ情報格納部7に登録されている i 番目のサポートしているエミュレーションの情報をリードする（ステップS262）。ここで、データが存在しない場合は、エミュ

レーション解析部11で調査した受信データで使用されているエミュレーションは、選択したサブプリンタではサポートされないため、サブプリンタがサポートしているエミュレーションの1番目の情報をリードし（ステップS263）、受信データをリードしたエミュレーションに変換した（ステップS264）後、データ送信部5に通知する（ステップS265）。

【0049】一方、ステップS262において、データが存在する場合には、リードしたデータと、受信データで使用したエミュレーションとの比較を行う（ステップS266）。比較の結果が一致していれば、受信データで使用しているエミュレーションは、選択したサブプリンタでサポートされるため、変換を行わずにデータ送信部5に通知する（ステップS265）。また、ステップS266で不一致の場合には $i = i + 1$ とし（ステップS267）、ステップS262に戻って次のサポートしているエミュレーションの情報をリードする。

【0050】尚、図12において、ステップS27～ステップS29の処理は、具体例1における図7のステップS15～ステップS17の処理と同様である。

【0051】〈効果〉以上のように具体例2によれば、具体例1の構成に加えて、自プリンタで印刷不可能な場合に印刷データを転送するサブプリンタのエミュレーションの情報を予め登録し、印刷不可能な場合は、サブプリンタのエミュレーションに変換して転送するようにしたので、ホストから印字先として指定したネットワークプリンタでエラーが発生し、印刷が不可能な場合でもそのネットワークプリンタの資源を最大限に利用し、効率よく印字ジョブを処理することができる。また、サブプリンタのエミュレーションが異なる場合でも印刷動作を行うことができる。

【0052】《具体例3》具体例3では、クライアントが印字先として指定したネットワークプリンタにおいて、何らかの原因で印字不可能なエラーが発生した場合、ネットワークプリンタは、ネットワーク上のファイルサーバを探し、そのサーバにテンポラリとして印刷データを保持するようにしたものである。

【0053】〈構成〉図14は、本発明のネットワークプリンタの具体例3を示す構成図である。図の装置は、コントローラ部1bとエンジン部2からなり、これらの全体としての機能は上記具体例1、2と同様である。

【0054】コントローラ部1bは、エミュレーション制御部3、データ受信部4、データ送信部5、MIBデータアクセス処理部6、ジョブ登録情報格納部13、印刷データ制御部14からなる。また、印刷データ制御部14は、ファイルサーバ検索部15、ファイル登録部16、ファイルロード部17からなる。

【0055】ここで、エミュレーション制御部3～MIBデータアクセス処理部6の機能は、上記各具体例と同様であるため、ここでの説明は省略する。一方、ジョブ

登録情報格納部13は、ファイルサーバにテンポラリとして作成した印刷データの情報を格納する機能部であり、例えば、次のような情報を格納する。

【0056】図15は、ジョブ登録情報格納部13の内容例を示す説明図である。図示例では、テンポラリデータ名とファイルサーバのMACアドレスをインデックスを付与して格納している。ここで、テンポラリデータとは印刷不可の状態から復旧した時点で読み出しを行うべきデータの名前であり、ファイルサーバのMACアドレスとは、読み出しの対象となるファイルサーバのMACアドレスを指している。

【0057】これらのデータの格納方法の一つとして、MIBにジョブ登録情報という独自のオブジェクトを定義し、プリンタの情報の一つとして保持する方法で実現する。例えば、図15の例では、データの項目として“テンポラリデータ名”と“ファイルサーバのMACアドレス”がある。例として、図示の状態では、エラーから復旧して最初に処理される印刷データのテンポラリデータ名は“DATA1”であり、そのデータを格納しているファイルサーバのMACアドレスは“00:80:87:XX:XX:XX”である。

【0058】図14に戻って、印刷データ制御部14は、印刷データに対する制御を行う機能部であり、ファイルサーバ検索部15は、テンポラリデータを作成すべきファイルサーバを検索する機能部、ファイル登録部16は、ファイルサーバへテンポラリデータを作成するための処理を行う機能部、ファイルロード部17は、エラーからの復旧後にファイルサーバから印刷データを読み出す機能部である。

【0059】〈動作〉 先ず、ジョブ登録情報格納部13の設定について説明する。図16は、ジョブ登録情報格納部13におけるジョブ登録情報格納処理の説明図である。図示のように、ネットワーク上を流れるデータ名とそのデータの宛先であるMACアドレスをそれぞれ登録順序のインデックスを付与し、MIBのデータ構造で格納する。

【0060】次に、印刷動作を説明する。図17は、自分宛のパケット受信時の印刷動作のフローチャートである。先ず、ホストから送信されたデータをデータ受信部4が受信バッファ4aに格納し（ステップS31）、印刷データ制御部14に通知する。これにより印刷データ制御部14は印刷データ制御処理を行い（ステップS32）、データが制御情報か印刷データかの判断処理を行う（ステップS33）。ステップS33において、印刷データでない場合はそのままエミュレーション制御部3に通知し（ステップS34）、印刷データであった場合は、エンジン部2が格納しているステータス情報2aが印刷状態か否かを判断する（ステップS35）。

【0061】プリンタのステータスが印刷可能状態の場合、印刷データ制御部14は印刷データをエミュレーション

ョン制御部3へ通知する。エミュレーション制御部3では、通知されたデータをエンジン部2にて印刷可能なフォーマットに変換し（ステップS36）、エンジン部2に通知する。エンジン部2は通知されたデータを元に用紙に印刷する（ステップS37）。

【0062】プリンタのステータスが印刷不可能状態の場合、印刷データ制御部14によって、ファイルサーバ検索部15が起動され、ファイルサーバ検索処理が行われる（ステップS38）。

【0063】図18は、ファイルサーバ検索処理のフローチャートである。ファイルサーバ検索部15は、ファイルサーバを検索するためのパケットを作成し、データ送信部5から送信する（ステップS381）。送信したパケットに対するファイルサーバからの応答を受信すると（ステップS382）、受信した応答パケットからファイルサーバのMACアドレスを取り出し、MIB（ジョブ登録情報格納部13）に格納する。

【0064】図19は、ファイルサーバ検索処理の説明図である。図示例は、ネットワーク上に複数のクライアントとファイルサーバ、ネットワークプリンタが存在するネットワークを示している。ここで、PR1は具体例3の構成を備えたネットワークプリンタであり、PR2はその他のプリンタを指している。また、PC1は印刷元パソコンであり、ネットワークプリンタPR1に対して印刷を行うホストである。FS1はネットワーク上に存在するファイルサーバである。

【0065】図20は、ファイルサーバ検索時に送受信される検索パケットと応答パケットの説明図である。プリンタからファイルサーバを検索するために送信される検索パケットは、情報フィールドにサーバのタイプについての情報を含み、ネットワーク上にブロードキャストされる（図19中の①に示す）。

【0066】検索パケットを受信したサーバの中で、検索パケットのサーバタイプと一致するファイルサーバは、その検索パケットに対する応答パケットをプリンタに対して送信する（図19中の②に示す）。この応答パケットには、ファイルサーバ自身のMACアドレスの情報が含まれている。そして、このような応答パケットを受信することによって、ネットワークプリンタPR1はファイルサーバの存在を認識することができる。

【0067】このような方法でサーバ検索が行える実例として、ディレクトリサービスのインターネット標準であるLDAP（Light Directory Access Protocol）が存在する。

【0068】図21に、LDAPサービスの概要を示す。このプロトコルは、X.500標準リンクであるDAP（Directory Access Protocol）を基礎としており、クライアントがアクセスすべき適切なファイルサーバを自動的に検索し、クライアントに対してファイル内の資源を提供することができる。この機能の実現には特定のネットワー

クOSは不要であり、サーバとクライアントに対してLD AP機能がインストールされていることのみ必須である。

【0069】図17に戻り、ファイルサーバ検索処理が行われると、ファイル登録部16が起動され、ファイル登録処理が行われる(ステップS39)。

【0070】図22は、ファイル登録処理のフローチャートである。ファイル登録部16は、対象となるファイルサーバ上に印刷データのテンポラリデータを作成し(ステップS391)、そのデータをMIB(ジョブ登録情報格納部13)に格納する(ステップS392)。この状態が図15に示す状態である。

【0071】また、このとき、プリンタに対してファイルサーバへのアクセス権が与えられていない場合、プリンタはファイルサーバにテンポラリデータを作成することができない。このような場合、予めファイルサーバ側に、プリンタがテンポラリデータ作成時に使用するログインアカウントを作成しておく必要がある。

【0072】次にエラー復旧時の動作を説明する。図23は、エラー復旧時のフローチャートである。プリンタがエラーから復旧すると、即ち、ステップS41において印刷可能になると、ファイルロード部17が起動され、ファイルロード処理が行われる(ステップS42)。

【0073】図24は、ファイルロード処理のフローチャートである。ファイルロード部17は、ジョブ登録情報格納部13のジョブ情報管理テーブルを参照し(ステップS421)、データが登録されていれば(ステップS422)、そのジョブ情報管理テーブルに登録されているMACアドレスを持つファイルサーバから、テーブルに登録されているデータ名を持つテンポラリデータをロードし(ステップS423)、エミュレーション制御部3に渡すことによって印刷を行う。

【0074】テンポラリデータのロードが終了すると、ファイルロード部17は、ファイルサーバ上からテンポラリデータを削除し(ステップS424)、ジョブ情報管理テーブル上から登録情報を削除する(ステップS425)。

【0075】〈効果〉以上のように、具体例3によれば、自プリンタで印刷不可能な場合は、ネットワーク上のファイルサーバを検索して、このファイルサーバにテンポラリデータを格納し、プリンタが復旧した場合はこのテンポラリデータを読み出して印刷を行うようにしたので、ホストから印字先として指定したネットワークプリンタでエラーが発生し、印刷が不可能な場合でもそのネットワークプリンタの資源を最大限に利用し、効率よく印字ジョブを処理することができる。また、印刷元のパソコンがネットワーク上のファイルサーバの存在を知らなくとも、プリンタ側で自動的にデータの保存や復旧時の印刷を行うことができる。

【0076】《具体例4》具体例4は、クライアントが

印字先として指定したネットワークプリンタにおいて、何らかの原因で印字不可能なエラーが発生した場合、それ以前に収集したネットワークの印刷履歴のログからユーザに最も近くに接続されていると推測される他のネットワークプリンタ(サブプリンタと呼ぶ)に再転送して印字するものである。

【0077】〈構成〉図25は、本発明のネットワークプリンタの具体例4を示す構成図である。図の装置は、コントローラ部1cとエンジン部2からなり、これらの全体としての機能は上記各具体例と同様である。

【0078】コントローラ部1cは、エミュレーション制御部3、データ受信部4、データ送信部5、MIBデータアクセス処理部6、印刷ログ情報格納部18、印刷ログ収集部19、印刷ログ解析部20からなる。

【0079】印刷ログ情報格納部18は、種々の印刷ログ情報を格納する格納部であり、例えば次のような情報を格納している。図26は、印刷ログ情報格納部18の内容例の説明図である。図示例では、ネットワーク上に存在するPCのアドレス、そのPCから印刷を行ったプリンタのMACアドレス、印刷データ形式、印刷回数といった情報からなる。これらのデータの格納方法の一つとして、MIBにジョブ登録情報という独自のオブジェクトを定義し、プリンタの情報の一つとして保持する方法で実現する。

【0080】印刷ログ情報格納部18の構造として、図示のように、一つのPCのMACアドレスに対するデータの項目として、“印刷先MACアドレス”、“印刷データ形式”、“印刷回数”がある。例えば、図示例において、PC1から印刷データとしてPCL形式のデータが送信される頻度が最も高いプリンタのMACアドレスは“00:80:87:XX:XX:XX”であり、本具体例の構成を備えたネットワークプリンタにおいて、印刷障害が発生している時にPC1からPCL形式のデータを受信すると、印刷データは“00:80:87:XX:XX:XX”のMACアドレスを持つプリンタに転送されることになる。

【0081】図25に戻って、印刷ログ収集部19は、ネットワーク上を流れるパケットから印刷ログを収集する機能を有している。印刷ログ解析部20は、印刷ログ収集部19にて収集した印刷ログを解析し、その解析結果をMIBデータアクセス処理部6に渡す機能を有すると共に、自プリンタが印刷不可能な状態である場合は、印刷ログ情報格納部18の印刷ログ情報に基づき、印刷データを転送すべきプリンタを決定する機能を有している。

【0082】〈動作〉図27は、パケット受信時のフローチャートである。プリンタはネットワーク上を流れるパケットを受信する度に印刷ログ収集処理を起動し(ステップS51)、その情報を、例えば具体例3のジョブ登録情報格納部の設定方法と同様な処理で、印刷ログ情報格納部18に設定する(ステップS52)。

【0083】図28は、印刷ログ収集処理の動作フローチャートである。印刷に使用されるパケットを受信する（ステップS521）と、印刷ログ解析部20は、パケットから送信元のMACアドレスを取り出し（ステップS522）、印刷ログ情報格納部18に存在しなければMIBデータアクセス処理部6によって登録を行う（ステップS523、S524）。

【0084】次に、印刷ログ解析部20は、送信先MACアドレスを取り出し（ステップS525）、プリンタ自身に割り当てられたアドレスと同一であるかを調べ（ステップS526）、同一でない場合は、印刷ログ情報格納部18の参照を行い、登録されていなければ新規に登録を行う（ステップS527、S528）。

【0085】そして、印刷データがどのような形式であるかを調べ（ステップS529）、対象となる印刷データ形式での印刷回数を1増加する（ステップS530）。

【0086】次に、自分宛の印刷データを受信した場合の動作を説明する。図29は、自分宛のパケットを受信した場合の動作フローチャートである。ここで、ステップS61～ステップS67までの処理は、具体例3の図17で示した自分宛のパケット受信時の動作におけるステップS31～ステップS37と同様である。

【0087】ステップS65において、印刷不可能であった場合、印刷ログ解析部20が起動し、印刷ログ解析処理を行う（ステップS68）。

【0088】図30は、印刷ログ解析処理のフローチャートである。プリンタのステータスが印刷不可能状態の場合、印刷ログ解析部20は受信した印刷データの送信元となるホストのMACアドレスにおいて、受信した印刷データ形式と同一の印刷データに対して最も印刷頻度が高いプリンタを印刷ログ情報格納部18の中で検索し、印刷データを転送すべきプリンタのMACアドレスを選択する。

【0089】即ち、印刷ログ情報中に印刷データ送信元が存在するかを調べ（ステップS681）、存在した場合は受信した印刷データの形式をチェックする（ステップS682）。そして、データ形式が一致し、かつ、印刷回数が最大の送信元MACアドレスを取り出し（ステップS683）、これが存在したら（ステップS684）、印刷データの再転送先として選択を行う（ステップS685）。

【0090】データ送信部5では、選択したサブプリンタのMACアドレスに対してデータを再送信する。

【0091】〈効果〉以上のように、具体例4によれば、自プリンタで印刷不可能な場合は、それ以前に収集したネットワークの印刷履歴のログから、データ形式が一致し、かつ、印刷回数が最も多いサブプリンタに再転送して印刷するようにしたので、ホストから印字先として指定したネットワークプリンタでエラーが発生し、印

刷が不可能な場合でもそのネットワークプリンタの資源を最大限に利用し、代替として最も適当と思われるネットワークプリンタに対して印刷ジョブを転送することによって、効率よく印字ジョブを処理することができる。

【0092】《具体例5》具体例5は、クライアントが印字先として指定したネットワークプリンタにおいて、何らかの原因で印字不可能なエラーが発生した場合、具体例4と同様にサブプリンタに再転送して印字すると共に、それ以前に収集したネットワークの電子メール送信ログから、印刷を行ったユーザの電子メールアドレスを調べ、サブプリンタに再転送したことを電子メールで通知するようにしたものである。

【0093】〈構成〉図31は、本発明のネットワークプリンタの具体例5を示す構成図である。図の装置は、コントローラ部1dとエンジン部2からなり、これらの全体としての機能は上記各具体例と同様である。

【0094】コントローラ部1dは、エミュレーション制御部3、データ受信部4、データ送信部5、MIBデータアクセス処理部6、印刷ログ・電子メールアドレス情報格納部18a、印刷ログ収集部19、印刷ログ解析部20、電子メール送信ログ収集部21、電子メール送信ログ解析部22からなる。

【0095】印刷ログ・電子メールアドレス情報格納部18aは、具体例4における種々の印刷ログ情報を格納すると共に、ユーザの電子メールアドレスを格納する格納部である。

【0096】図32は、印刷ログ・電子メールアドレス情報格納部18aの内容例の説明図である。図示例では、ネットワーク上に存在するPCのアドレス、そのPCから印刷を行ったプリンタのMACアドレス、印刷データ形式、印刷回数、印刷を行ったPCから送信された電子メールアドレスといった情報からなる。これらのデータの格納方法の一つとして、MIBにジョブ登録情報という独自のオブジェクトを定義し、プリンタの情報の一つとして保持する方法で実現する。

【0097】印刷ログ・電子メールアドレス情報格納部18aの構造として、図示のように、一つのPCのMACアドレスに対するデータの項目として、“印刷先MACアドレス”、“印刷データ形式”、“印刷回数”、“電子メールアドレス”がある。例えば、図示例において、PC1から印刷データとしてPCL形式のデータが送信される頻度が最も高いプリンタのMACアドレスは“00:80:87:XX:XX:XX”であり、その電子メールアドレスは“xxx@xxx.xx.xx”である。そのため、本具体例の構成を備えたネットワークプリンタにおいて、印刷障害が発生している時にPC1からPCL形式のデータを受信すると、印刷データは“00:80:87:XX:XX:XX”のMACアドレスを持つプリンタに転送され、転送先のプリンタのMACアドレス情報を含んだ電子メールが、“xxx@xxx.xx.xx”に送信されることになる。

【0098】また、電子メール送信ログ収集部21は、ネットワーク上を流れる電子メール送信用のパケットを収集して電子メールアドレスの情報を収集する機能を有している。更に、電子メール送信ログ解析部22は、電子メール送信ログ収集部21で収集した電子メール送信用のパケットからそのログを解析し、これをMIBデータアクセス処理部6に渡す機能を有すると共に、印刷不可能時、サブプリンタにデータが再転送された場合は、印刷ログ・電子メールアドレス情報格納部18aに情報に基づき、そのサブプリンタについての情報を印刷要求元のPCの電子メールアドレスに送信する機能を有している。その他の各構成は具体例4と同様であるため、ここでの説明は省略する。

【0099】〈動作〉図33は、パケット受信時のフローチャートである。プリンタはネットワーク上を流れるパケットを受信する度に電子メールログ収集処理を起動し（ステップS71）、その情報を、例えば具体例3のジョブ登録情報格納部の設定方法と同様な処理で、印刷ログ・電子メールアドレス情報格納部18aに設定する（ステップS72）。

【0100】図34は、電子メール送信ログ収集処理の動作フローチャートである。電子メール送信に使用されるパケットを受信する（ステップS721）と、電子メール送信ログ解析部22は、そのパケットから送信元のMACアドレスを取り出す（ステップS722）。次に、印刷ログ・電子メールアドレス情報格納部18aの印刷元のMACアドレスを確認し（ステップS723）、印刷元のMACアドレスとして存在するかを調べ（ステップS724）、存在しなければ廃棄する。

【0101】ステップS724において、印刷元MACアドレスとして存在した場合は、パケットから送信元電子メールアドレスを取り出し（ステップS725）、印刷ログ・電子メールアドレス情報格納部18aに登録する（ステップS726）。

【0102】次に、自分宛の印刷データを受信した場合の動作を説明する。図35は、自分宛のパケットを受信した場合の動作フローチャートである。ここで、ステップS81～ステップS89までの処理は、具体例4の図29で示したステップS61～ステップS69と同様であるため、ここでの説明は省略する。

【0103】ステップS89でデータ送信を行うと、次に電子メール送信ログ解析処理を行う（ステップS90）。

【0104】図36は、電子メール送信ログ解析処理のフローチャートである。データ送信部5で選択したサブプリンタのMACアドレスに対してデータが再送信されると、電子メール送信ログ解析部22は、印刷ログ・電子メールアドレス情報格納部18aを参照し（ステップS901）、印刷元の電子メールアドレスが登録されているかどうかを確認する（ステップS902）。電子メ

ールアドレスが登録されていない場合はそのまま終了するが、登録されている場合は、印刷データが転送されたサブプリンタのMACアドレスを取り出す（ステップS903）。そして、取り出したMACアドレスの情報を含んだ電子メールを作成し（ステップS904）、登録されている電子メールアドレスに送信する（ステップS905）。

【0105】〈効果〉以上のように、具体例5によれば、具体例4の構成に加えて、自プリンタで印刷不可能な場合は、それ以前に収集したネットワークの電子メールの送信ログから、印刷元の電子メールを調べ、この印刷元に対して電子メールで通知するようにしたので、ホストから印字先として指定したネットワークプリンタでエラーが発生し、印刷が不可能な場合でもそのネットワークプリンタの資源を最大限に利用し、効率よく印字ジョブを処理することができる。また、ユーザがどのネットワークプリンタで印刷が行われているかを的確に把握することができる。

【0106】《具体例6》具体例6は、クライアントが印字先として指定したネットワークプリンタにおいて、何らかの原因で印字不可能なエラーが発生した場合、具体例4と同様にサブプリンタに再転送して印字すると共に、印刷を行ったユーザに対してポップアップ・メッセージにより、サブプリンタに再転送したことを通知するようにしたものである。

【0107】尚、ポップアップ・メッセージとは、ネットワーク上のパソコン間でメッセージのやり取りを行う機能であり、メッセージを送った場合に相手側のパソコンの画面上にメッセージがポップアップするよう構成されている機能を指す。

【0108】〈構成〉図37は、本発明のネットワークプリンタの具体例6を示す構成図である。図の装置は、コントローラ部1eとエンジン部2からなり、これらの全体としての機能は上記各具体例と同様である。

【0109】コントローラ部1eは、エミュレーション制御部3、データ受信部4、データ送信部5、MIBデータアクセス処理部6、印刷ログ情報格納部18b、印刷ログ収集部19、印刷ログ解析部20、ポップアップ・メッセージ送信部23からなる。

【0110】印刷ログ情報格納部18bは、具体例4における種々の印刷ログ情報を格納する機能を有している。

【0111】図38は、印刷ログ情報格納部18bの内容例の説明図である。図示例では、ネットワーク上に存在するPCのアドレス、そのPCから印刷を行ったプリンタのMACアドレス、印刷データ形式、印刷回数、使用プロトコルからなる。これらのデータの格納方法の一つとして、MIBにジョブ登録情報という独自のオブジェクトを定義し、プリンタの情報の一つとして保持する方法で実現する。

【0112】例として、図38に示した印刷ログ情報格納部18bにおいて、MACアドレスが“00:10:11:XX:XX:XX”であるPCIから印刷データとしてPCLデータが送信される頻度が元も高いプリンタのMACアドレスは、“00:80:87:XX:XX:XX”である。また、PCIには、TCP/IPとIPXの少なくとも二つのプロトコルがインストールされており、TCP/IPを使用している印刷が先に行われたことを示している。そのため、本具体例の構成を備えたネットワークプリンタにおいて、印刷障害が発生している時にPCIからPCL形式のデータを受信すると、印刷データは“00:80:87:XX:XX:XX”のMACアドレスを持つプリンタに転送され、転送先のプリンタのMACアドレス情報を含んだポップアップ・メッセージが、TCP/IPプロトコルを使用して“00:10:11:XX:XX:XX”のMACアドレスに送信されることになる。

【0113】また、ポップアップ・メッセージ送信部23は、印刷データを再転送したプリンタについての情報をポップアップ・メッセージとして送信する機能を有している。その他の各構成は具体例4と同様であるため、ここでの説明は省略する。

【0114】また、印刷元PCのシステム部にはプリンタドライバのインストールと共に、ポップアップ・メッセージ受信部がインストールされている。

【0115】図39は、印刷元PCの構成図である。図において、100は印刷元PCを示し、ポップアップ・メッセージ受信部101とポップアップ・メッセージ制御部102を備えている。ポップアップ・メッセージ受信部101は、ネットワークプリンタから、ポップアップ・メッセージを含んだパケットを受け取った場合、そのパケットからポップアップ・メッセージを取り出し、これをポップアップ・メッセージ制御部102に渡す機能を有している。

【0116】ポップアップ・メッセージ制御部102は、例えば、Windowsの標準機能として備えられているポップアップサービスの制御部であり、ポップアップ・メッセージ表示処理部103とポップアップ・メッセージ表示部104を備えている。

【0117】ポップアップ・メッセージ表示処理部103は、他のパソコンやポップアップ・メッセージ受信部101からポップアップ・メッセージを受け取った場合に、そのメッセージをポップアップ・メッセージ表示部104に表示する機能を有している。また、ポップアップ・メッセージ表示部104は、例えば、ディスプレイからなり、ポップアップ・メッセージを表示する表示部である。

【0118】〈動作〉図40は、パケット受信時のフローチャートである。プリンタはネットワーク上を流れるパケットを受信する(ステップS101)度に印刷ログ収集処理および印刷ログ収集処理2を起動し(ステップS102、S103)、その情報を、例えば具体例3の

ジョブ登録情報格納部の設定方法と同様な処理で、印刷ログ情報格納部18bに設定する。ここで、印刷ログ収集処理は、具体例4の図27におけるステップS52の印刷ログ収集処理と同様である。また、印刷ログ収集処理2は、印刷に使用されるパケットのフォーマットから使用しているプロトコルを調べる処理である。

【0119】図41は、印刷ログ収集処理2の動作フローチャートである。電子メール送信に使用されるパケットを受信する(ステップS1031)と、そのパケットのフォーマットから使用しているプロトコルを調べる(ステップS1032)。次に、印刷ログ情報格納部18bの使用プロトコルを確認し(ステップS1033)、登録されているかを調べ(ステップS1034)、登録されていれば廃棄し、登録されていなければ使用プロトコルとして印刷ログ情報格納部18bに登録する(ステップS1035)。

【0120】次に、自分宛の印刷データを受信した場合の動作を説明する。図42は、自分宛のパケットを受信した場合の動作フローチャートである。ここで、ステップS1101～ステップS1109までの処理は、具体例4の図29で示したステップS61～ステップS69と同様であるため、ここでの説明は省略する。

【0121】ステップS1109でデータ送信を行うと、次にポップアップ・メッセージ送信処理を行う(ステップS1110)。

【0122】図43は、ポップアップ・メッセージ送信処理のフローチャートである。データ送信部5で選択したサブプリンタのMACアドレスに対してデータが再送信されると、ポップアップ・メッセージ送信部23は、印刷ログ情報格納部18bを参照し(ステップS11101)、印刷データが転送されたサブプリンタのMACアドレスと、印刷元PCが印刷に使用したプロトコルとして最初に登録されたプロトコル情報を取り出す(ステップS11102、S11103)。

【0123】そして、取り出されたプロトコル情報に従ってパケットを作成し(ステップS11104)、取り出したMACアドレスの情報を含んだポップアップ・メッセージと共に、印刷データ送信元のMACアドレスへパケットを送信する(ステップS11105、S11106)。

【0124】次に、ポップアップ・メッセージを受信した印刷元PCの処理を説明する。図44は、印刷元PCにおけるポップアップ・メッセージ受信処理のフローチャートである。印刷元PC100のポップアップ・メッセージ受信部101は、プリンタからのパケットを受信すると(ステップS1201)、ポップアップ・メッセージを取り出し(ステップS1202)、ポップアップ・メッセージ表示処理部103をコールし、そのメッセージを渡す(ステップS1203)。ポップアップ・メッセージ表示処理部103は、受け取ったメッセージを

ポップアップ・メッセージ表示部 104 に表示する。

【0125】〈効果〉以上のように、具体例 6 によれば、具体例 4 の構成に加えて、自プリンタで印刷不可能な場合は、印刷元 PC に対してポップアップ・メッセージで通知するようにしたので、ホストから印字先として指定したネットワークプリンタでエラーが発生し、印刷が不可能な場合でもそのネットワークプリンタの資源を最大限に利用し、効率よく印字ジョブを処理することができる。また、ネットワークで使用しているプロトコルや印刷を行ったユーザに対するメールアドレス割り当ての有無にかかわらず、実際に印刷を行ったプリンタをユーザに通知することが可能となり、印刷結果を探す手間が省けるという効果がある。

【0126】尚、上記具体例 6 では、ポップアップ・メッセージ制御部として Windows のポップアップサービスを用いたが、これに限定されるものではなく、同様のポップアップ・メッセージの送受信処理が行えるものであれば同様に使用可能である。

【0127】また、上記具体例 1～6 において、サブプリンタ情報格納部 7 やジョブ登録情報格納部 13 あるいは印刷ログ情報格納部 18 b 等の情報設定に SNMP を用いたが、SNMP の他にも “Web ブラウザ” や “Telnet” “プリンタのオペレーションパネル” 等が利用可能である。

【図面の簡単な説明】

【図 1】本発明のネットワークプリンタの具体例 1 を示す構成図である。

【図 2】サブプリンタ情報格納部の内容例を示す説明図である。

【図 3】ネットワークプリンタの動作環境を示す説明図である。

【図 4】サブプリンタ情報格納部の情報設定動作のフローチャートである。

【図 5】ホストからのサブプリンタ情報設定要求の説明図である。

【図 6】プリンタにおけるサブプリンタ情報格納部への情報格納処理の説明図である。

【図 7】具体例 1 の印刷動作時のフローチャートである。

【図 8】プロトコル変換の動作フローチャートである。

【図 9】プロトコル変換の説明図である。

【図 10】印刷不可時の動作説明図である。

【図 11】本発明のネットワークプリンタの具体例 2 を示す構成図である。

【図 12】具体例 2 の印刷動作時のフローチャートである。

【図 13】エミュレーション変換部の動作フローチャートである。

【図 14】本発明のネットワークプリンタの具体例 3 を示す構成図である。

【図 15】ジョブ登録情報格納部の内容例を示す説明図である。

【図 16】ジョブ登録情報格納部におけるジョブ登録情報格納処理の説明図である。

【図 17】具体例 3 の自分宛の packets 受信時の印刷動作のフローチャートである。

【図 18】ファイルサーバ検索処理のフローチャートである。

【図 19】ファイルサーバ検索処理の説明図である。

【図 20】ファイルサーバ検索時に送受信される検索 packets と応答 packets の説明図である。

【図 21】LDAP サービスの概要を示す説明図である。

【図 22】ファイル登録処理のフローチャートである。

【図 23】エラー復旧時のフローチャートである。

【図 24】ファイルロード処理のフローチャートである。

【図 25】本発明のネットワークプリンタの具体例 4 を示す構成図である。

【図 26】印刷ログ情報格納部の内容例の説明図である。

【図 27】packets 受信時のフローチャートである。

【図 28】印刷ログ収集処理の動作フローチャートである。

【図 29】自分宛の packets を受信した場合の動作フローチャートである。

【図 30】印刷ログ解析処理のフローチャートである。

【図 31】本発明のネットワークプリンタの具体例 5 を示す構成図である。

【図 32】印刷ログ・電子メールアドレス情報格納部の内容例の説明図である。

【図 33】packets 受信時のフローチャートである。

【図 34】電子メール送信ログ収集処理の動作フローチャートである。

【図 35】自分宛の packets を受信した場合の動作フローチャートである。

【図 36】電子メール送信ログ解析処理のフローチャートである。

【図 37】本発明のネットワークプリンタの具体例 6 を示す構成図である。

【図 38】印刷ログ情報格納部の内容例の説明図である。

【図 39】印刷元 PC の構成図である。

【図 40】packets 受信時のフローチャートである。

【図 41】印刷ログ収集処理 2 の動作フローチャートである。

【図 42】自分宛の packets を受信した場合の動作フローチャートである。

【図 43】ポップアップ・メッセージ送信処理のフローチャートである。

【図 44】印刷元 PC におけるポップアップ・メッセー

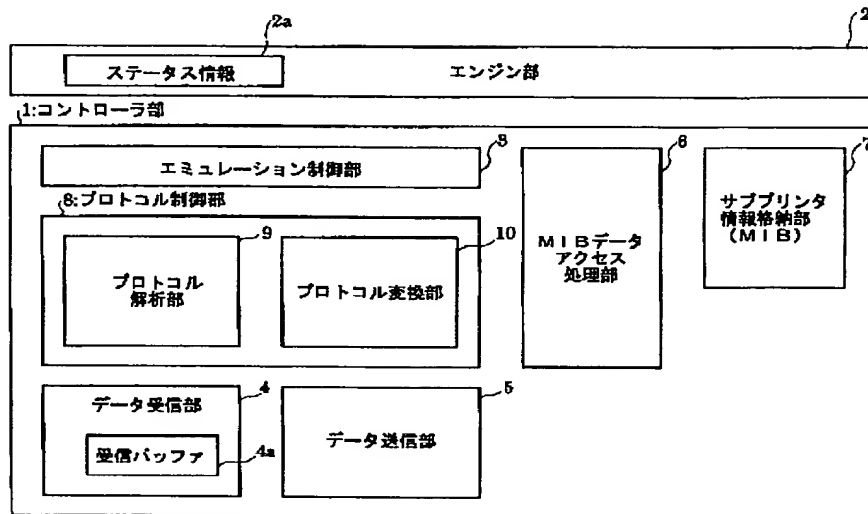
ジ受信処理のフローチャートである。

【符号の説明】

- 3 エミュレーション制御部
- 5 データ送信部
- 7 サブプリンタ情報格納部
- 8 プロトコル制御部
- 15 ファイルサーバ検索部
- 17 ファイルロード部

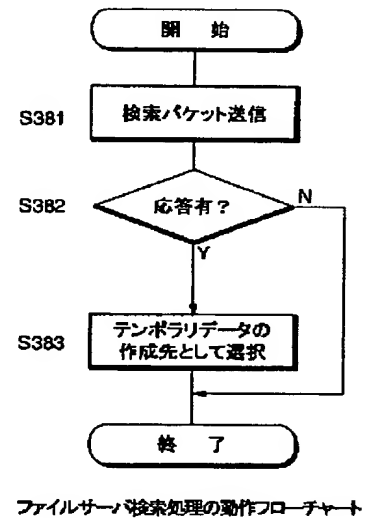
- 19 印刷ログ収集部
- 20 印刷ログ解析部
- 21 電子メール送信ログ収集部
- 22 電子メール送信ログ解析部
- 23 ポップアップ・メッセージ送信部
- 101 ポップアップ・メッセージ受信部
- 102 ポップアップ・メッセージ制御部

【図 1】



具体例 1 の構成図

【図 18】



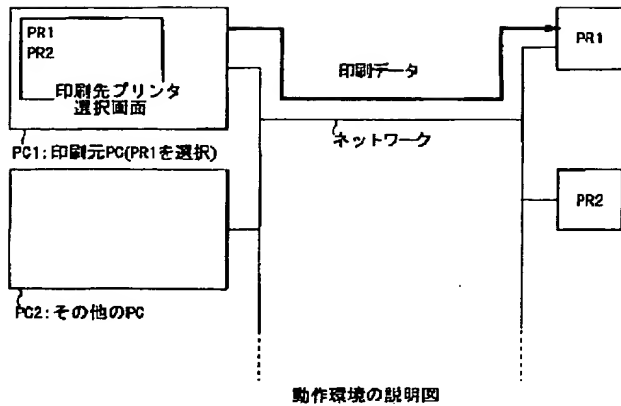
ファイルサーバ検索処理の動作フローチャート

【図 2】

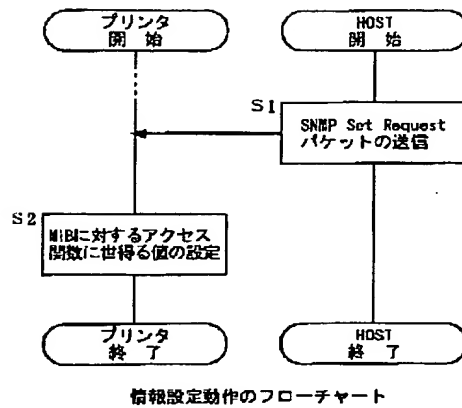
優先順位	プリンタ名	MACアドレス	サポートしている プロトコル	サポートしている エミュレーション
1	PR2	00:80:87:XX:XX:XX	1,LPR 2,FTP	1,PCL 2,IBM PPR 3,EPSON FX
2	PR3	XX:XX:XX:XX:XX:XX	1,R-Printer 2,Q-Server	1,PCL 2,Hiper Windows
...
9	PR10	XX:XX:XX:XX:XX:XX	1,EtherTalk	1,PCL 2,PS

サブプリンタ情報格納部の内容例の説明図

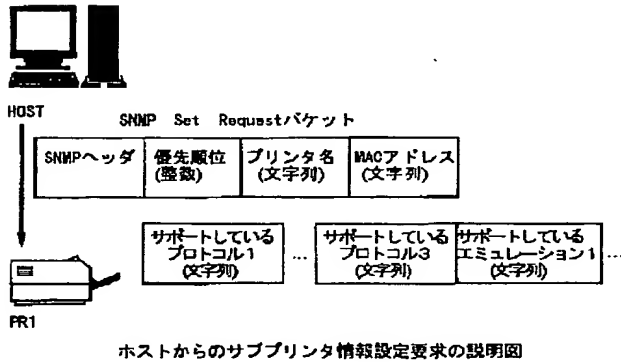
【図 3】



【図 4】



【図 5】

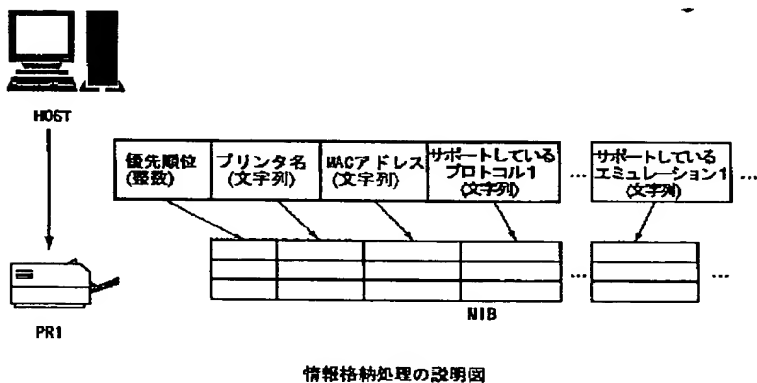


【図 15】

INDEX	テンポラリデータ名	ファイルサーバのMACアドレス
1	DATA1	00:80:87:XX:XX:XX
2	DATA2	XX:XX:XX:XX:XX:XX
...
9	DATA9	XX:XX:XX:XX:XX:XX

ジョブ登録情報格納部の内容例の説明図

【図 6】



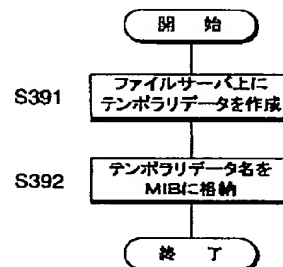
【図 16】

登録順序 (整数)	データ名 (文字列)	MACアドレス (文字列)

MIB

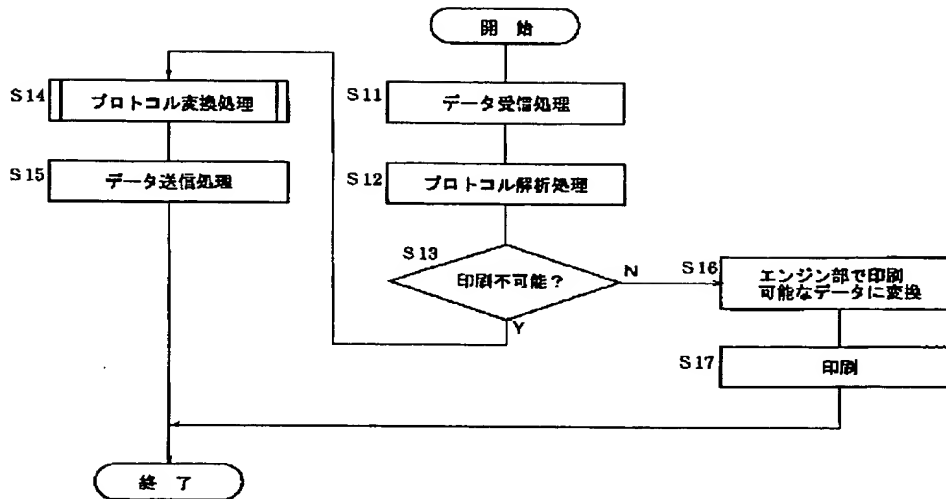
プリンタにおけるジョブ登録情報格納処理の説明図

【図 22】

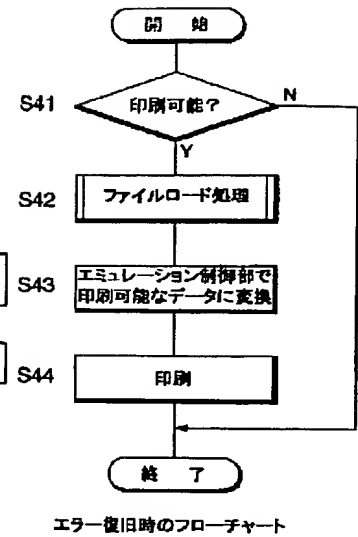


ファイル登録処理の動作フローチャート

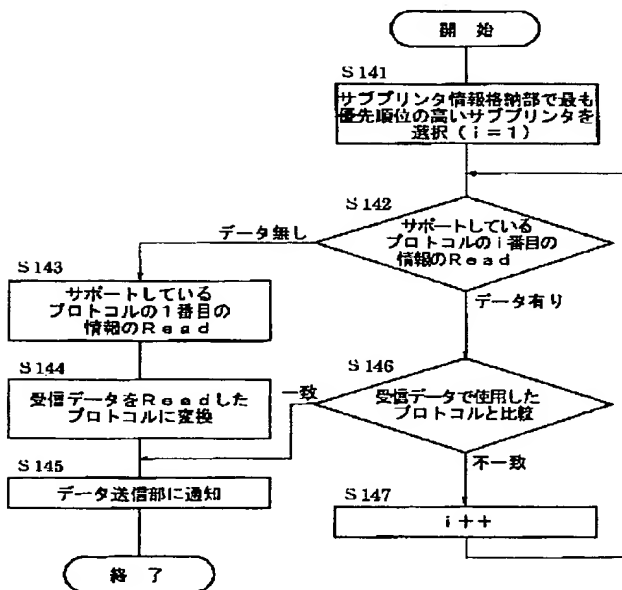
【図 7】



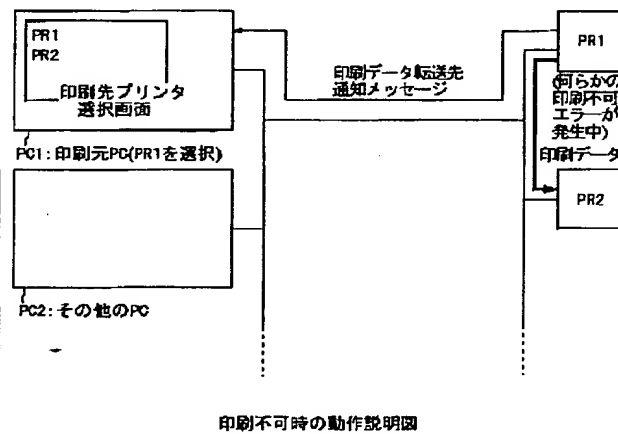
【図 23】



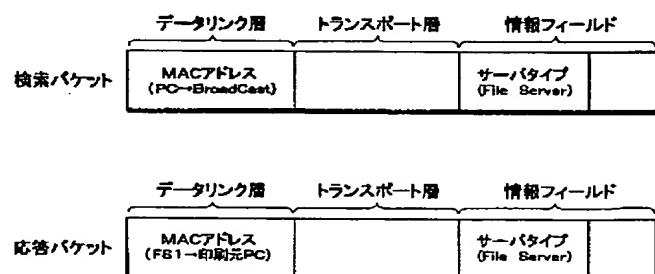
【図 8】



【図 10】

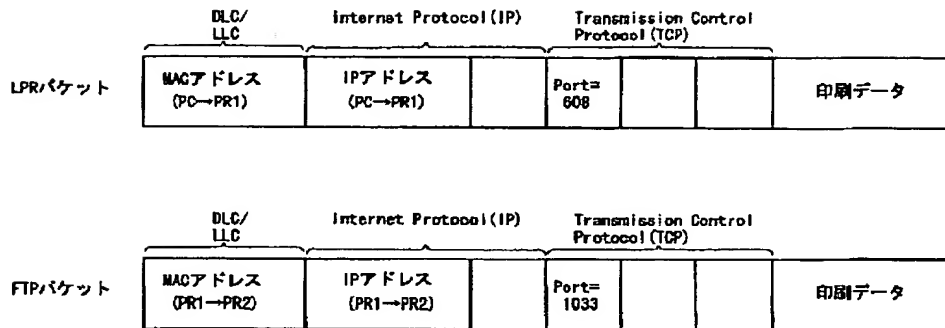


【図 20】



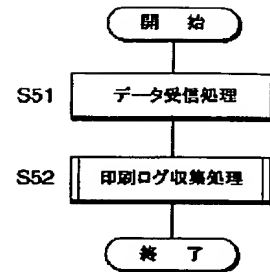
検索パケットと応答パケットの説明図

【図 9】



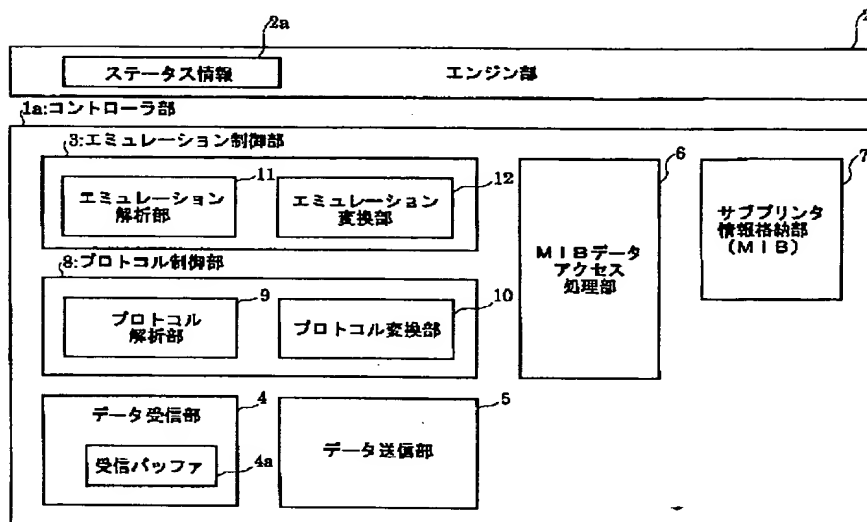
プロトコル変換の説明図

【図 27】



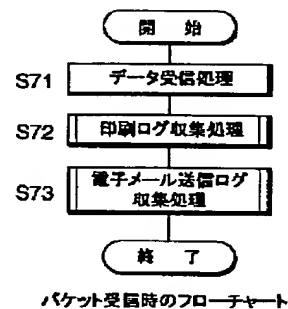
パケット受信時のフローチャート

【図 11】



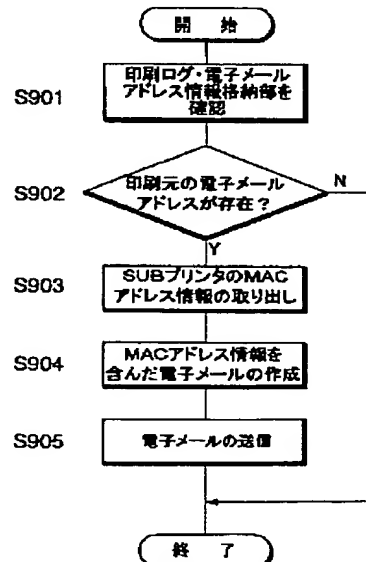
具体例 2 の構成図

【図 33】



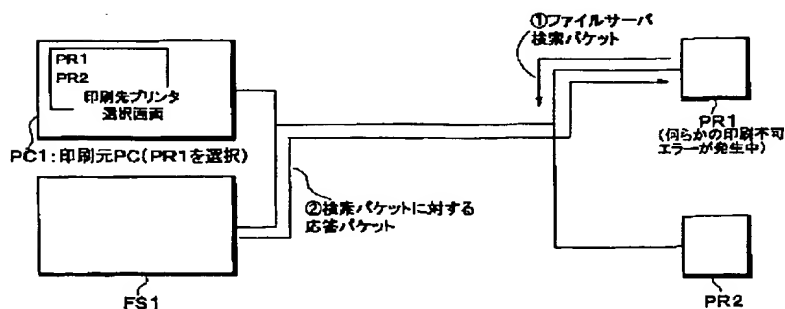
パケット受信時のフローチャート

【図 36】



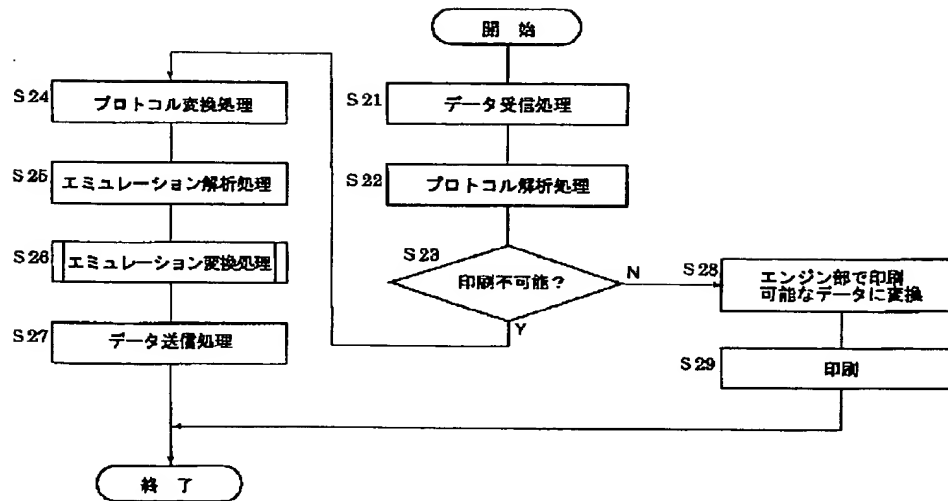
電子メール送信ログ解析処理の動作フローチャート

【図 19】



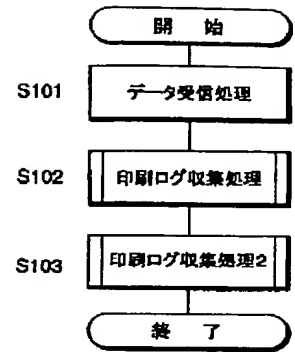
ファイルサーバ検索処理の説明図

【図12】



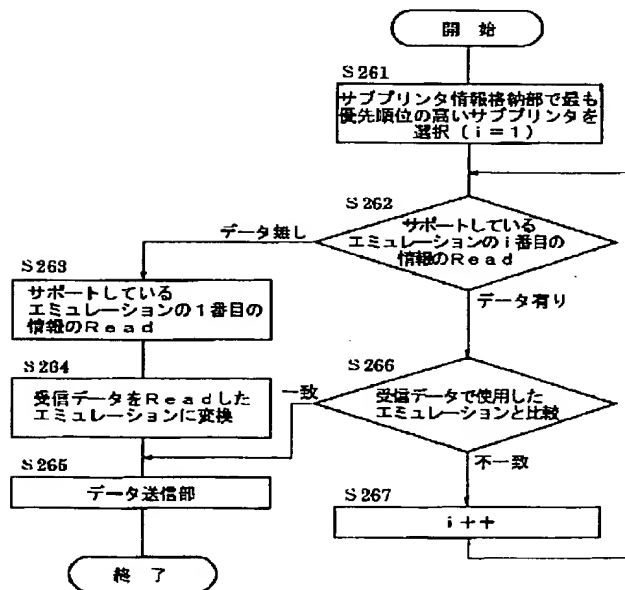
印刷動作時のフローチャート

【図40】



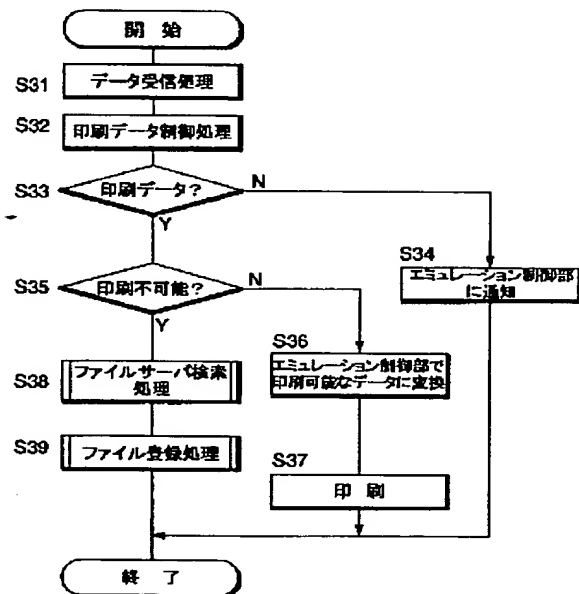
パケット受信時のフローチャート

【図13】



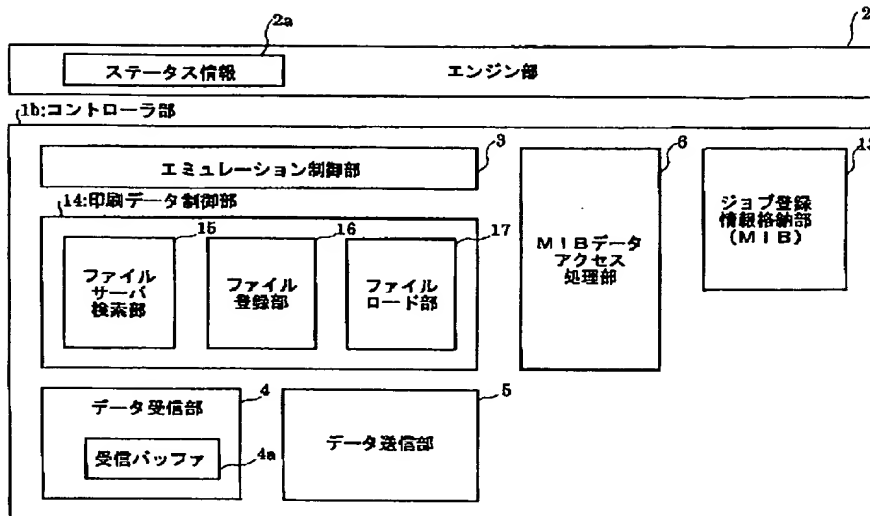
エミュレーション変換部の動作フローチャート

【図17】



自分宛のパケット受信時の印刷動作フローチャート

【図 14】

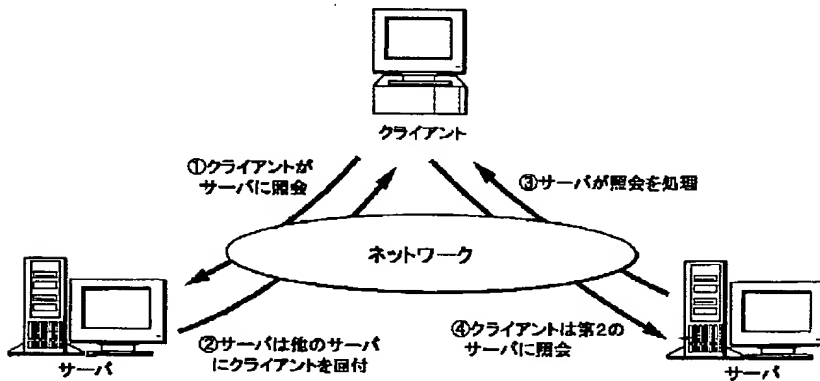


具体例 3 の構成図

【図 43】



【図 21】



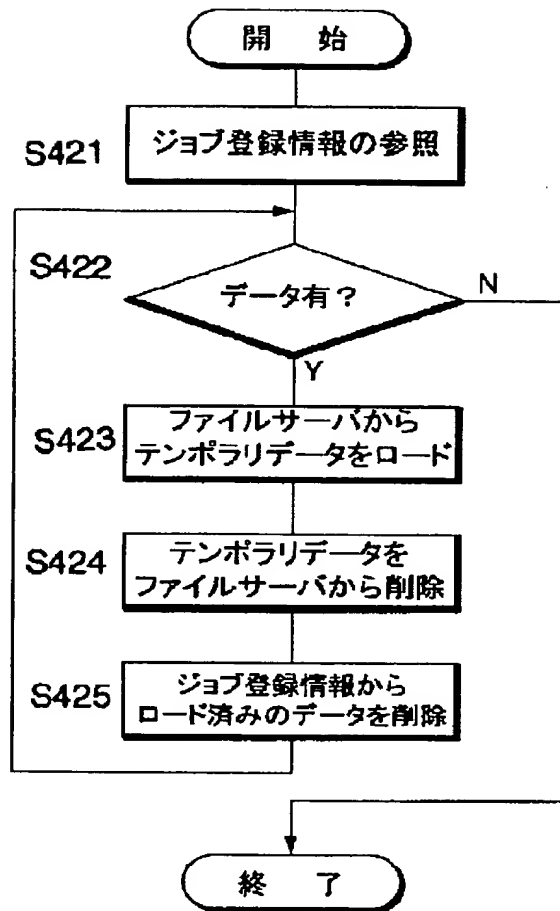
LDAPサービスの概要の説明図

【図 26】

Index	印刷元のMACアドレス	印刷先のMACアドレス (複数)	印刷データ形式	印刷回数
1	00:11:11:XX:XX:XX	(1)-00:80:87:XX:XX:XX (2)-00:00:00:XX:XX:XX (3)-00:00:01:XX:XX:XX (4)-00:00:02:XX:XX:XX	(1)-PCL (2)-PCL (3)-PostScript (4)-不明	(1)-10 (2)-4 (3)-9 (4)-2
2	00:22:22:XX:XX:XX	(1)-00:00:01:XX:XX:XX (2)-00:80:87:XX:XX:XX	(1)-PCL (2)-PostScript	(1)-7 (2)-12
...
9	00:99:99:XX:XX:XX	(1)-XX:XX:XX:XX:XX:XX	(1)-不明	(1)-1

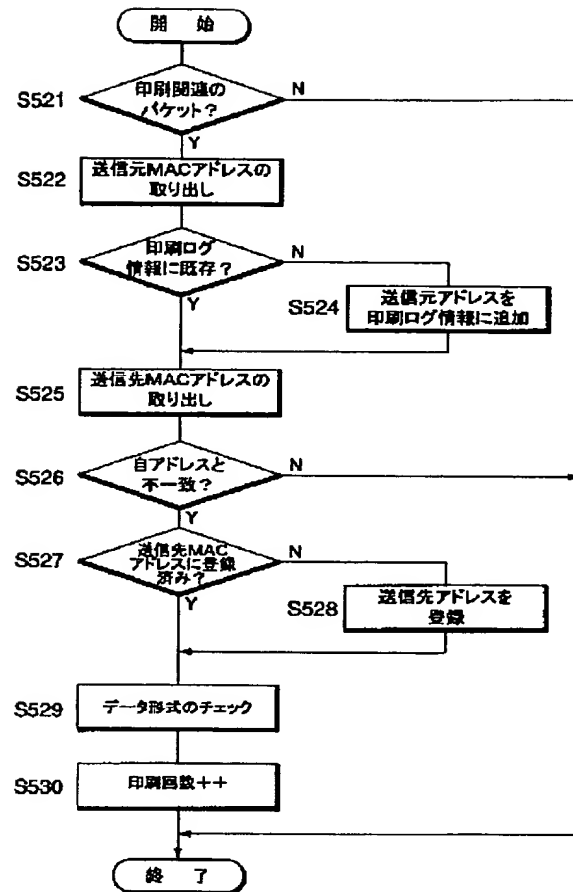
印刷ログ情報格納部の内容例の説明図

【図 24】



ファイルロード処理の動作フローチャート

【図 28】



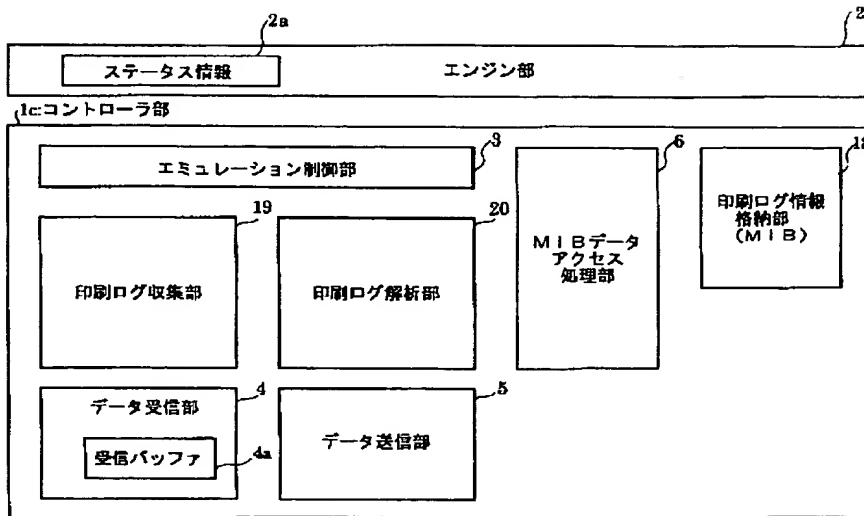
印刷ログ収集処理の動作フローチャート

【図 32】

Index	印刷元のMACアドレス	印刷先のMACアドレス (複数)	印刷データ形式	印刷回数	電子メールアドレス
1	00:11:11:XX:XX:XX	(1)-00:80:87:XX:XX:XX (2)-00:00:00:XX:XX:XX (3)-00:00:01:XX:XX:XX (4)-00:00:02:XX:XX:XX	(1)-PCL (2)-PCL (3)-PS (4)-不明	(1)-10 (2)-4 (3)-9 (4)-2	xxx@xxx.xxx
2	00:22:22:XX:XX:XX	(1)-00:00:01:XX:XX:XX (2)-00:80:87:XX:XX:XX	(1)-PCL (2)-PS	(1)-7 (2)-12	yyy@yyy.yy.yy
...
9	00:99:99:XX:XX:XX	(1)-XX:XX:XX:XX:XX:XX	(1)-不明	(1)-1	(なし)

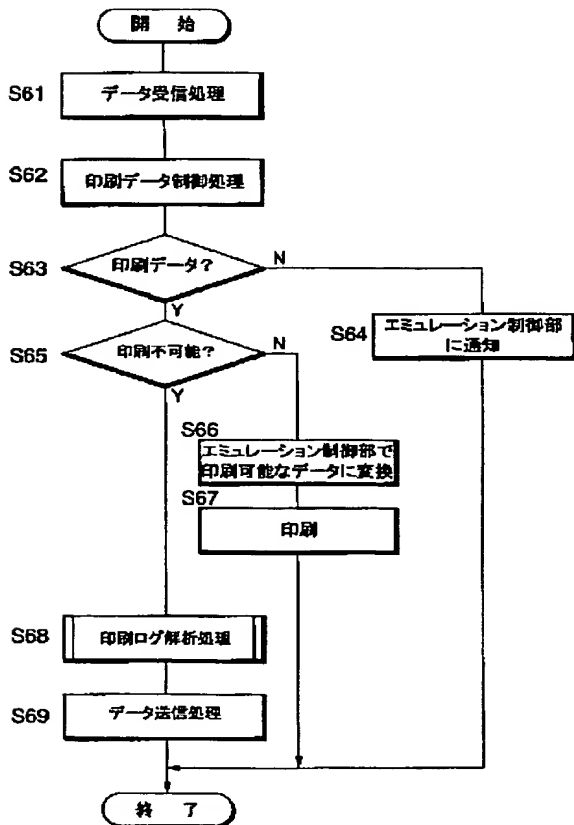
印刷ログ・電子メールアドレス情報格納部の内容例の説明図

【図 25】



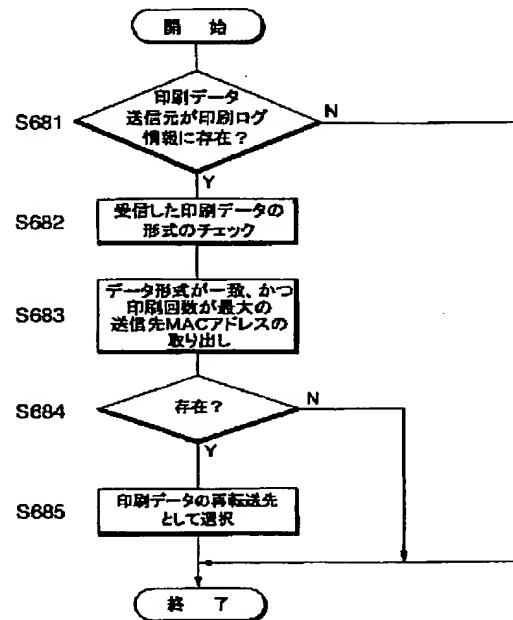
具体例 4 の構成図

【図 29】



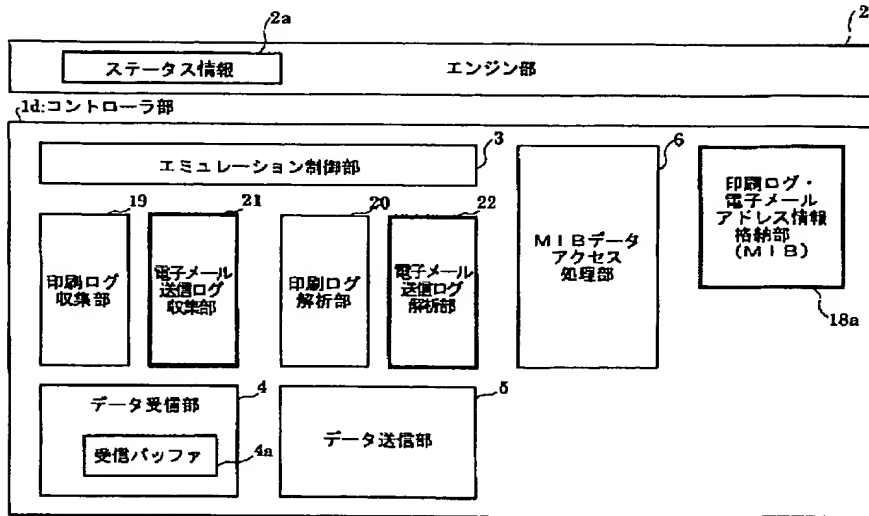
自分宛のバケット受信時の動作フローチャート

【図 30】



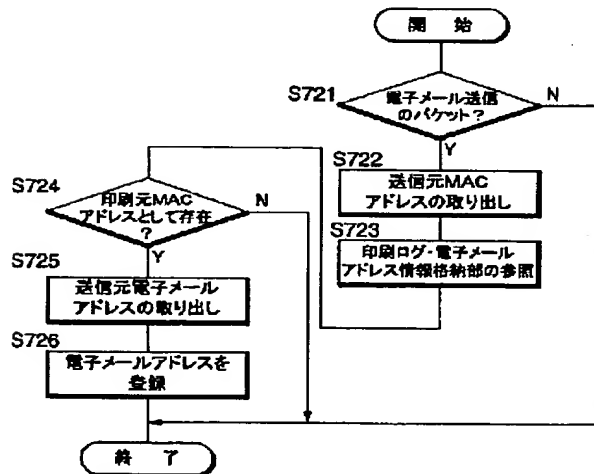
印刷ログ解析処理の動作フローチャート

【図 3 1】



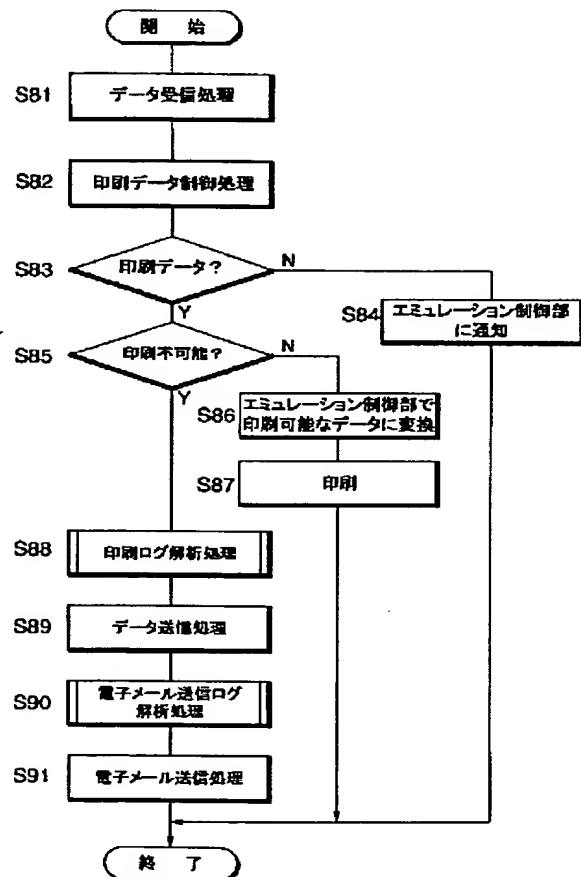
具体例 5 の構成図

【図 3 4】



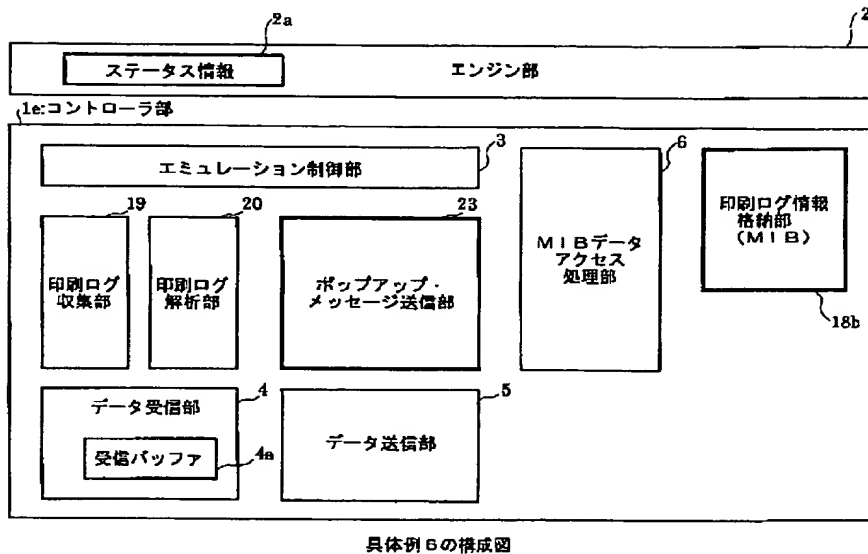
電子メール送信ログ収集処理の動作フローチャート

【図 3 5】



自分宛のバケットを受信した場合の動作フローチャート

【図 37】

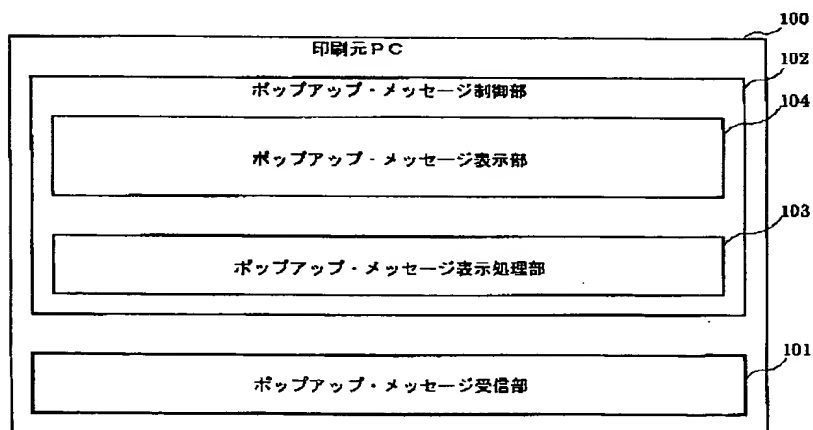


【図 38】

Index	印刷元のMACアドレス	印刷先のMACアドレス (複数)	印刷データ形式	印刷回数	使用プロトコル
1	00:11:11:XX:XX:XX	(1)-00:80:87:XX:XX:XX (2)-00:00:00:XX:XX:XX (3)-00:00:01:XX:XX:XX (4)-00:00:02:XX:XX:XX	(1)-PCL (2)-PCL (3)-PostScript (4)-不明	(1)-10 (2)-4 (3)-9 (4)-2	(1)IPX (2)TCP/IP
2	00:22:22:XX:XX:XX	(1)-00:00:01:XX:XX:XX (2)-00:80:87:XX:XX:XX	(1)-PCL (2)-PostScript	(1)-7 (2)-12	(1)TCP/IP
...
9	00:99:99:XX:XX:XX	(1)-XX:XX:XX:XX:XX:XX	(1)-不明	(1)-1	(1)EtherTalk

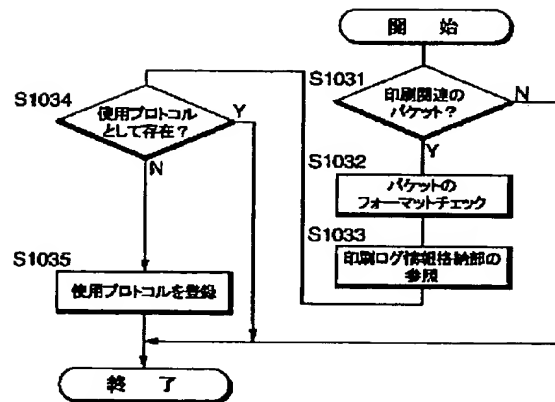
印刷ログ情報格納部の内容例の説明図

【図 39】

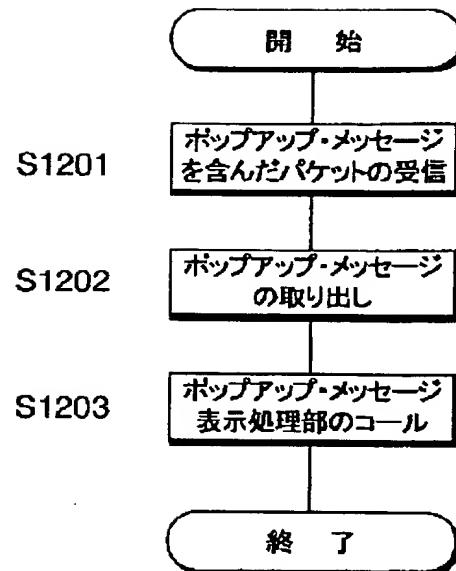


印刷元PCの構成図

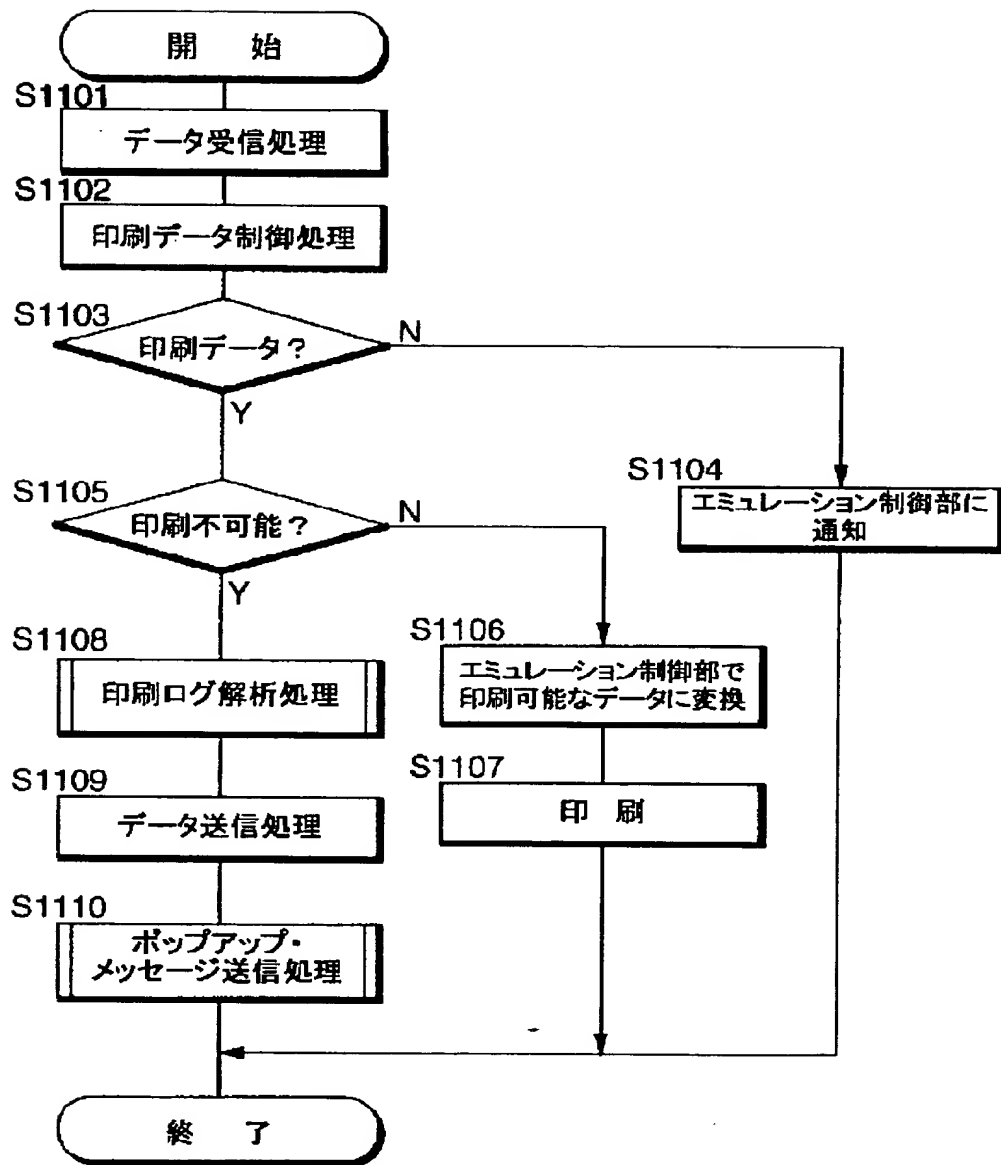
【図 4 1】



【図 4 4】



【図 42】



自分宛のパケットを受信した場合の動作フローチャート